In the Specification:

Please substitute the following paragraphs for the corresponding paragraphs beginning at the indicated locations in the specification as originally filed.

Page 1, line 6+:

This application is related to U. S. Patent Applications ~~09/___,___~~ 09/736,444, now U. S. Patent 6,757,439 (Attorney's Docket RAL-99-0177), and ~~09/___,___~~ 09/736,445, now U. S. Patent 6,373,412 (Attorney's Docket END9-2000-0113US1), both filed December 15, 2000, entitled JPEG Packed Block Structure and Fast JPEG Huffman Coding and Decoding, respectively, and U. S. Patent Application 09/896,117, entitled Faster Lossless Rotation of JPEG Images, filed concurrently herewith, all of which are assigned to the assignee of the present application and hereby fully incorporated by reference.

Page 9, line 2+:

The foregoing and other objects, aspects and advantages will be better understood from the following detailed description of a preferred embodiment of the invention with reference to the drawings, in which:

Figure 1 is a schematic depiction of a data format for digital quantized transform coefficients compatible with the JPEG standard,

Figure 2 is a schematic depiction of a packed block data format as disclosed in the above-incorporated U. S. Patent application ~~09/___,___~~ 09/736,445, ~~(Attorney's docket No. END9-2000-0113US1)~~ (U. S. Patent 6,373,412),

Figure 3 is a schematic depiction of an improved packed block data format in accordance with the present invention,

Figure 4 is a schematic depiction of a simplified form of an improved packed block data format in

accordance with the invention, and

Figure 5 is a schematic depiction of an alternative form of the improved packed block data format in accordance with the invention.

Page 12, line 5+:

Referring now to Figure 2, a JPEG compatible packed block format as disclosed in the above-incorporated application ~~09/___,___~~ 09/736,444, now U. S. Patent 6,757,439 (JPEG Packed Block Structure~~,~~ ~~Attorney's docket RAL-99-0177~~) is shown. Each DCT block starts with a length byte 21 followed by two bytes 22 to save the location of the EOB for sequential coding and the current EOB for progressive coding. The quantized DC coefficient 23 is stored in the next two bytes/sixteen bits. This may be represented as either the actual DC coefficient value or the difference in DC coefficient from the previous block (e.g. as a prediction). However, the former is generally preferred when further DCT domain processing is anticipated to render the blocks (and their order) self-contained and independent of each other. On the other hand, where the purpose of the format is to save the information while statistics are being accumulated (e.g. for generating custom Huffman tables, saving the DC coefficient value as the prediction (not shown in Figure 1) converted into an S byte followed by the one or two bytes of extra bits will minimize later processing. It is possible to use both formats simultaneously.

Page 12, line 29+:

Each non-zero AC coefficient is stored in two or more bytes. The first byte 24 is the R/S byte used for Huffman encoding, (i.e. the high order nibble R = four bits) equals the run of zero-valued AC coefficients in zig-zag order up to fifteen and the low order nibble S

= four bits) is the number of extra bits necessary to uniquely code the non-zero magnitude. A preferred form of this packed format stores the extra bits in the next one or two bytes (e.g. E1 or E1 and E2) 25, depending on whether or not the second byte is needed (i.e. S>8). That is, E2 is an optional second byte which is only needed if S>8. The EOB byte is used if EOB1<64. Since the ZRLs and E2 are data dependent, data is ~~access~~ accessed one byte at a time. An alternative implementation always follows the R/S byte with the actual AC coefficient value in two bytes. The final byte is the symbol 0x00 which indicates that an EOB is to be coded. ZRL is a byte 27 of the form 0xF0 used when the run of zero coefficients is greater than 15.